

**IN THE CLAIMS**

Please amend the claims as provided below.

1. (Original) A method for generating a cubic Bezier triangular control mesh corresponding to a triangular primitive, comprising:

receiving vertex parameters corresponding to three vertices of the triangular primitive, wherein the vertex parameters for each vertex includes three-dimensional coordinates and a normal vector;

calculating two control points corresponding to each edge of three edges of the triangular primitive based on the vertex parameters of vertices that define the edge; and

calculating a central control point using the vertex parameters for each of the three vertices and the control points corresponding to the three edges.

2. (Original) The method of claim 1, wherein calculating two control points corresponding to an edge further comprises:

mapping a segment equal to a fraction of a length of the edge to a first plane defined by a first normal corresponding to a first vertex of the vertices that define the edge, wherein the segment is mapped such that the segment is coplanar with the edge and the first normal, wherein a first end of the segment as mapped corresponds to the first vertex and wherein a second end of the segment as mapped defines a first control point corresponding to the edge; and

mapping the segment to a second plane defined by a second normal corresponding to a second vertex of the vertices that define the edge, wherein the segment is mapped such that the segment is coplanar with the edge and the second normal, wherein a first end of the segment as mapped corresponds to the second vertex and wherein a second end of the segment as mapped defines a second control point corresponding to the edge.

3. (Original) The method of claim 2, wherein the fraction of the length is within a range of one-quarter to one-half of the length.

4. (Original) The method of claim 3, wherein the fraction of the length is approximately one-third of the length.

5. (Original) The method of claim 2, wherein the fraction of the length is specified by a user.

6. (Original) The method of claim 1, wherein calculating two control points corresponding to an edge further comprises:

*BD*  
*Cmt*  
projecting a second vertex of the vertices that define the edge into a first plane defined by a first normal corresponding to a first vertex of the vertices that define the edge to produce a first reference segment corresponding to the edge, wherein projecting the second vertex is performed in a direction parallel to the first normal, wherein a fraction of the first reference segment defines a first control point corresponding to the edge; and

projecting the first vertex into a second plane defined by a second normal corresponding to the second vertex to produce a second reference segment corresponding to the edge, wherein projecting the first vertex is performed in a direction parallel to the second normal, wherein a fraction of the second reference segment defines a second control point corresponding to the edge.

7. (Original) The method of claim 6, wherein the fraction of the first reference segment is approximately one-third of the first reference segment, and wherein the fraction of the second reference segment is approximately one-third of the second reference segment.

8. (Original) The method of claim 1, wherein calculating the central control point further comprises:

reflecting each of the three vertices through a line defined by a pair of the control points to produce a reflected point, wherein each control point of the pair of control points for reflection of a particular vertex is determined using a plane defined by the normal corresponding to the particular vertex, wherein the reflected point is defined by a set of three-dimensional coordinates;

averaging the three-dimensional coordinates of the reflected points produced by reflecting the three vertices to produce coordinates corresponding to the central control point.

9. (Original) The method of claim 1, wherein calculating the central control point further comprises determining each coordinate value for the central control point by:

adding corresponding coordinate values of the control points for each of the edges to produce a first sum;

adding corresponding coordinate values for the three vertices to produce a second sum;

dividing the first sum by four to produce a first value;

dividing the second sum by six to produce a second value; and

subtracting the second value from the first value the coordinate value.

10. (Original) The method of claim 1, wherein calculating the central control point further comprises combining at least a portion of the vertex parameters of the three vertices and parameters for the control points corresponding to the edges based on a user-specified combining parameters.

11. (Original) A method for generating planar video graphics primitives for representing a non-planar surface, comprising:

receiving vertex parameters corresponding to three vertices of a triangular primitive that represents the non-planar surface, wherein the vertex parameters for each vertex include three-dimensional coordinates and a normal vector;

calculating a set of control points corresponding to the triangle primitive based on the three vertices, wherein the set of control points and the vertices define a cubic Bezier triangular control mesh; and

generating a plurality of planar triangle primitives using the cubic Bezier triangular control mesh, wherein the plurality of planar triangle primitives approximate the non-planar surface in three dimensions.

12. (Original) The method of claim 11 further comprises processing the plurality of planar triangle primitives using a three dimensional pipeline to produce pixel data corresponding to an image that represents the non-planar surface.

13. (Original) The method of claim 11, wherein calculating the set of control points includes:

calculating two control points corresponding to each edge of three edges of the triangular primitive based on the vertex parameters of vertices that define the edge; and

calculating a central control point using the vertex parameters for each of the three vertices and the control points corresponding to the three edges.

14. (Original) The method of claim 13 wherein calculating two control points corresponding to an edge further comprises:

mapping a segment equal to a fraction of a length of the edge to a first plane defined by a first normal corresponding to a first vertex of the vertices that define the edge, wherein the segment is mapped such that the segment is coplanar with the edge and the first

normal, wherein a first end of the segment as mapped corresponds to the first vertex and wherein a second end of the segment as mapped defines a first control point corresponding to the edge; and

mapping the segment to a second plane defined by a second normal corresponding to a second vertex of the vertices that define the edge, wherein the segment is mapped such that the segment is coplanar with the edge and the second normal, wherein a first end of the segment as mapped corresponds to the second vertex and wherein a second end of the segment as mapped defines a second control point corresponding to the edge.

15. (Original) The method of claim 14, wherein the fraction of the length is within a range of one-quarter to one-half of the length.

16. (Original) The method of claim 15, wherein the fraction of the length is approximately one-third of the length.

17. (Original) The method of claim 13, wherein calculating two control points corresponding to an edge further comprises:

projecting a second vertex of the vertices that define the edge into a first plane defined by a first normal corresponding to a first vertex of the vertices that define the edge to produce a first reference segment corresponding to the edge, wherein projecting the second vertex is performed in a direction parallel to the first normal, wherein a fraction of the first reference segment defines a first control point corresponding to the edge; and

projecting the first vertex into a second plane defined by a second normal corresponding to the second vertex to produce a second reference segment corresponding to the edge, wherein projecting the first vertex is performed in a direction parallel to the second normal,

wherein a fraction of the second reference segment defines a second control point corresponding to the edge.

18. (Original) The method of claim 17, wherein the fraction of the first reference segment is approximately one-third of the first reference segment, and wherein the fraction of the second reference segment is approximately one-third of the second reference segment.

19. (Original) The method of claim 13, wherein calculating the central control point further comprises:

*AD1*  
*CMT*  
reflecting each of the three vertices through a line defined by a pair of the control points to produce a reflected point, wherein each control point of the pair of control points for reflection of a particular vertex is determined using a plane defined by the normal corresponding to the particular vertex, wherein the reflected point is defined by a set of three-dimensional coordinates;

averaging the three-dimensional coordinates of the reflected points produced by reflecting the three vertices to produce coordinates corresponding to the central control point.

20. (Original) The method of claim 13, wherein calculating the central control point further comprises determining each coordinate value for the central control point by:

adding corresponding coordinate values of the control points for each of the edges to produce a first sum;

adding corresponding coordinate values for the three vertices to produce a second sum;

dividing the first sum by four to produce a first value;

dividing the second sum by six to produce a second value; and

subtracting the second value from the first value the coordinate value.

21. (Original) The method of claim 13, wherein calculating the central control point further comprises combining at least a portion of the vertex parameters of the three vertices and parameters for the control points corresponding to the three edges based on a user-specified combining parameters.

22. (Original) The method of claim 11, wherein generating the plurality of planar triangle primitive is based on a tessellation level, wherein the tessellation level determines a number of planar triangle primitives included in the plurality of planar triangle primitives.

AD1  
CMT  
23. (Original) The method of claim 11, wherein generating the plurality of planar triangle primitives further comprises determining vertex values for vertices of each planar triangle primitive of the plurality of planar triangle primitives, wherein at least a portion of the vertex values are determined by evaluating a Bernstein polynomial using the set of control points and the vertices that define the cubic Bezier triangular control mesh.

24. (Original) The method of claim 23, wherein evaluating the Bernstein polynomial further comprises evaluating the cubic Bezier triangular control mesh at the vertices of each planar triangle primitive of the plurality of planar triangle primitives.

25. (Original) The method of claim 24, wherein evaluating the Bernstein polynomial is based on calculations using Barycentric coordinates, with respect to the triangular primitive, of the vertices of each planar triangle primitive of the plurality of planar triangle primitives.

26. (Original) The method of claim 23, wherein the vertex values for each vertex of each planar triangle primitive of the plurality of planar triangle primitives further includes a planar primitive vertex normal corresponding to each vertex of the planar triangle primitive, wherein each planar primitive vertex normal is generated by interpolation using vertex normals of the triangular primitive.

27. (Original) The method of claim 26, wherein each planar primitive vertex normal is re-normalized after generation using interpolation.

28. (Original) The method of claim 26, wherein the interpolation used in generation of each planar primitive vertex normal uses Barycentric coordinates.

29. (Original) The method of claim 26, wherein the interpolation is linear interpolation.

30. (Original) The method of claim 26, wherein the interpolation is quadratic interpolation.

31. (Original) The method of claim 23, wherein the vertex values for each vertex of each planar triangle primitive of the plurality of planar triangle primitives include at least one of texture coordinates, color data, and fog data.

32. (Original) The method of claim 23, wherein at least a portion of the vertex values for each vertex of each planar triangle primitive of the plurality of planar triangle primitives is calculated using linear interpolation.

33. (Original) A video graphics processing system, comprising:  
a processor that generates a higher-order graphics primitive;  
a control point generation block operably coupled to the processor, wherein the control point generation block generates a control mesh for the higher-order graphics primitive;  
a tessellation block operably coupled to the control point generation block, wherein the tessellation block tessellates the higher-order graphics primitive to produce a plurality of planar triangle primitives; and



a three-dimensional graphics pipeline operably coupled to the tessellation block, wherein the three-dimensional graphics pipeline processes the plurality of planar triangle primitives to produce pixel data.

34. (Original) The video graphics processing system of claim 33, wherein the higher-order graphics primitive is a higher-order triangular primitive that is defined by vertex parameters for three vertices, wherein the vertex parameters for each vertex include a normal vector and three-dimensional coordinates.

35. (Original) The video graphics processing system of claim 34, wherein the control point generation block generates the control mesh by:

calculating two control points corresponding to each edge of three edges of the triangular primitive based on the vertex parameters of vertices that define the edge; and

calculating a central control point using the vertex parameters for each the three vertices and the control points corresponding to the three edges.

36. (Original) The video graphics processing system of claim 35, wherein the control point generation block calculates the two control points corresponding to an edge by:

mapping a segment equal to a fraction of a length of the edge to a first plane defined by a first normal corresponding to a first vertex of the vertices that define the edge, wherein the segment is mapped such that the segment is coplanar with the edge and the first normal, wherein a first end of the segment as mapped corresponds to the first vertex and wherein a second end of the segment as mapped defines a first control point corresponding to the edge; and

mapping the segment to a second plane defined by a second normal corresponding to a second vertex of the vertices that define the edge, wherein the segment is mapped such that the segment is coplanar with the edge and the second normal, wherein a first end of the segment as

mapped corresponds to the second vertex and wherein a second end of the segment as mapped defines a second control point corresponding to the edge.

37. (Original) The video graphics processing system of claim 36, wherein the fraction of the length is within a range of one-quarter to one-half of the length.

38. (Original) The video graphics processing system of claim 37, wherein the fraction of the length is approximately one-third of the length.

39. (Original) The video graphics processing system of claim 36, wherein the fraction of the length is specified by a user.

40. (Original) The video graphics processing system of claim 35, wherein the control point generation block calculates the two control points corresponding to an edge by:

projecting a second vertex of the vertices that define the edge into a first plane defined by a first normal corresponding to a first vertex of the vertices that define the edge to produce a first reference segment corresponding to the edge, wherein projecting the second vertex is performed in a direction parallel to the first normal, wherein a fraction of the first reference segment defines a first control point corresponding to the edge; and

projecting the first vertex into a second plane defined by a second normal corresponding to the second vertex to produce a second reference segment corresponding to the edge, wherein projecting the first vertex is performed in a direction parallel to the second normal, wherein a fraction of the second reference segment defines a second control point corresponding to the edge.

41. (Original) The video graphics processing system of claim 40, wherein the fraction of the first reference segment is approximately one-third of the first reference segment, and

wherein the fraction of the second reference segment is approximately one-third of the second reference segment.

42. (Original) The video graphics processing system of claim 35, wherein the control point generation block calculates the central control point by:

reflecting each of the three vertices through a line defined by a pair of the control points to produce a reflected point, wherein each control point of the pair of control points for reflection of a particular vertex is determined using a plane defined by the normal corresponding to the particular vertex, wherein the reflected point is defined by a set of three-dimensional coordinates;

averaging the three-dimensional coordinates of the reflected points produced by reflecting the three vertices to produce coordinates corresponding to the central control point.

43. (Original) The video graphics processing system of claim 33 further comprises at least one vector engine operably coupled to the control point generation block, wherein the at least one vector engine is used for performing calculations associated with generating the control mesh.

44. (Original) The video graphics processing system of claim 33 further comprises memory operably coupled to the control point generation block and the tessellation block, wherein the memory stores the control mesh for use by the tessellation block.

45. (Original) The video graphics processing system of claim 33, wherein the tessellation block produces the plurality of planar triangle primitives using at least one of linear interpolation and calculations using Barycentric coordinates.

46. (Original) The video graphics processing system of claim 33, wherein the control point generation block calculates the central control point by combining at least a portion of the

vertex parameters of the three vertices and parameters for the control points corresponding to the edges based on a user-specified combining parameters.

47. (Original) The video graphics processing system of claim 33, wherein the control point generation block calculates the central control point such that it includes determining each coordinate value for the central control point by:

adding corresponding coordinate values of the control points for each of the edges to produce a first sum;

adding corresponding coordinate values for the three vertices to produce a second sum;

dividing the first sum by four to produce a first value;

dividing the second sum by six to produce a second value; and

subtracting the second value from the first value the coordinate value.

48. (New) A method for generating a cubic Bezier triangular control mesh corresponding to a triangular primitive, comprising:

receiving vertex parameters corresponding to three vertices of the triangular primitive, wherein the vertex parameters for each vertex includes three-dimensional coordinates and a normal vector;

calculating two scalar control points corresponding to each edge of three edges of the triangular primitive based on the vertex parameters of vertices that define the edge; and

calculating a central control using the vertex parameters for each of the three vertices and the scalar control points using the vertex parameters for each of the three vertices and the control points corresponding to the three edges.

49. (New) The method of claim 48 including calculating the central control point using a dot product of the vertex parameters for each of the three vertices and the scalar control points corresponding to the three edges.

